Apprentissage automatique

Méthodes à noyaux - motivation

MODÉLISATION NON-LINÉAIRE

Sujets: prédicteur non-linéaire

- On a vu plusieurs algorithmes qui produisent des modèles linéaires (régression ou classification)

- Malheureusement, pas tous les problèmes peuvent être résolus avec un modèle linéaire

- Par contre, on peut obtenir des modèles non-linéaires à l'aide de fonctions de base non-linéaires

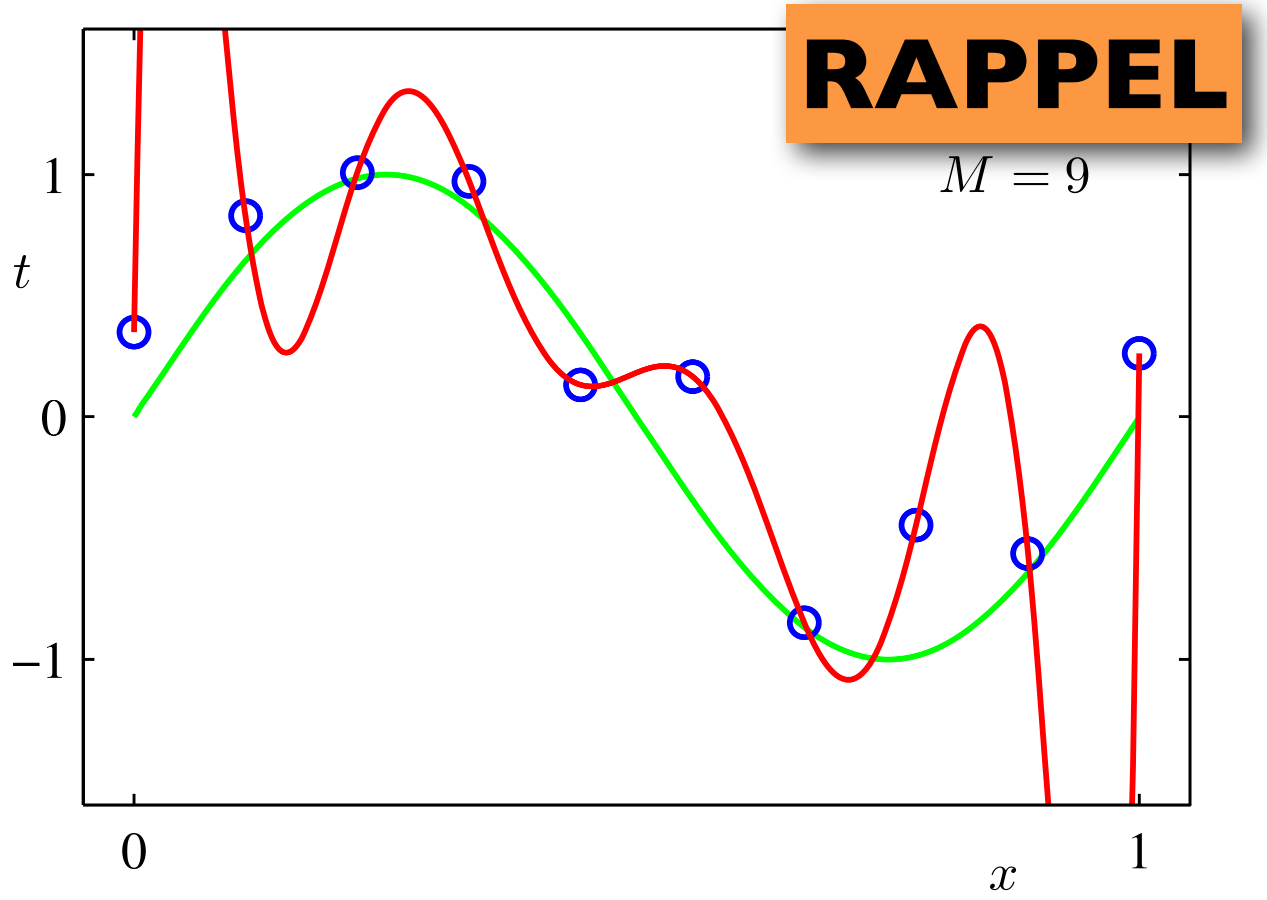
FONCTION DE BASE

Sujets: fonctions de base polynomiales

- Exemple : fonctions de bases polynomiales ($1D$)

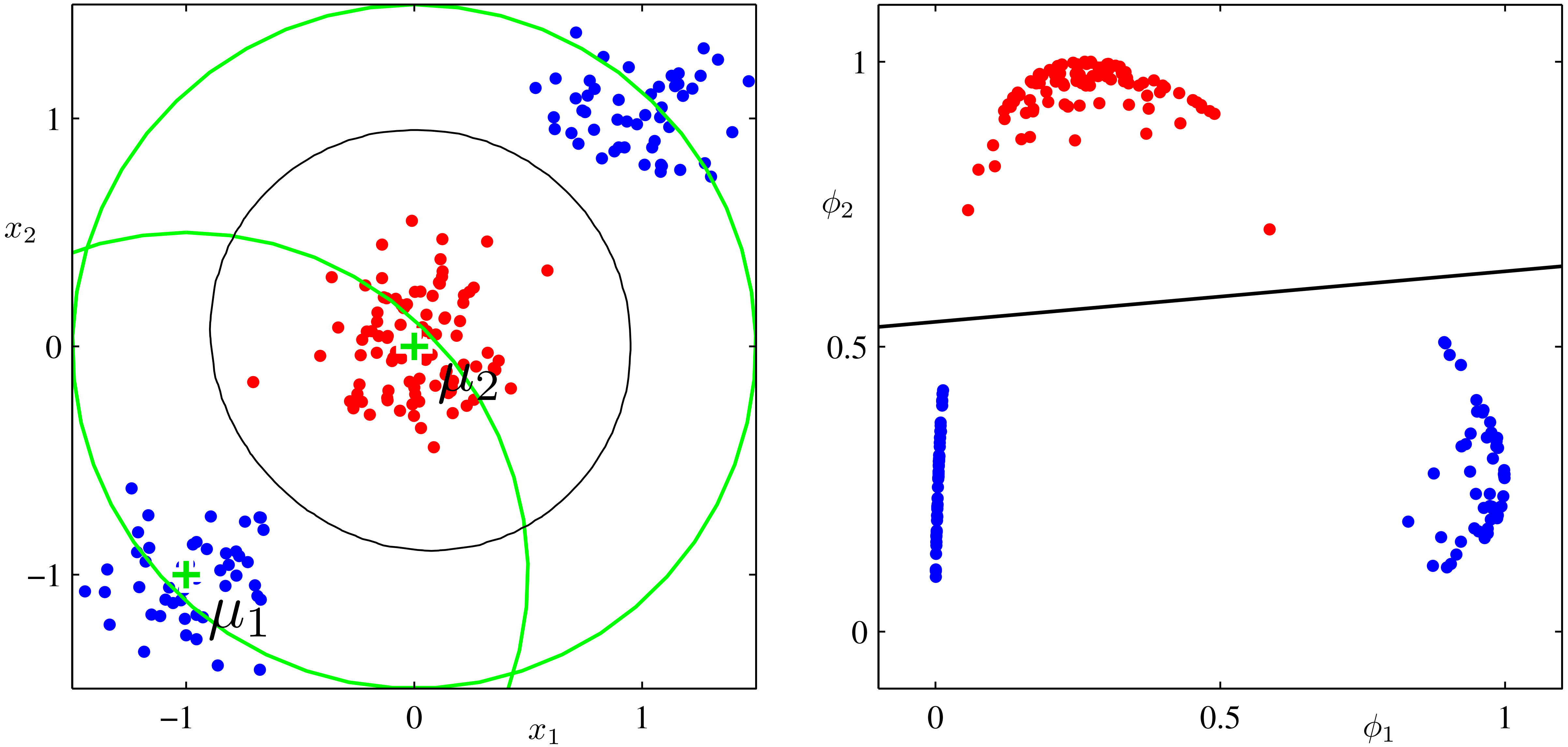
$$\phi_j(x) = x^j$$

- On retrouve alors la régression polynomiale



Approche Probabiliste Discriminante

Sujets: fonctions de bases



espace représenté par $\mathbf{x}$ — espace représenté par ϕ

RAPPEL

2 fonctions de bases gaussiennes ϕ_j

$$\exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

MÉTHODES À NOYAUX

Sujets: noyau, méthodes à noyaux

- On va maintenant voir une façon très simple d'introduire des fonctions de bases non-linéaires dans un modèle linéaire
  - les fonctions de bases vont être définies implicitement (pas besoin de représenter explicitement en mémoire $\phi(\mathbf{x})$!)
  - on aura seulement à calculer une comparaison $k(\mathbf{x}, \mathbf{x}')$ entre les entrées $\mathbf{x}$ et $\mathbf{x}'$
- La fonction $k(\mathbf{x}, \mathbf{x}')$ est appelée un **noyau**
  - les algorithmes utilisant un noyau sont appelées **méthodes à noyaux**

Apprentissage automatique

Méthodes à noyaux - représentation duale (régression)

RÉGRESSION

Sujets: régression

- Revenons au problème de la régression (régularisée) :

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) - t_n\right\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

- Si on fixe le gradient par rapport à $\mathbf{w}$ à 0, on observe que

$$\mathbf{w} = -\frac{1}{\lambda}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) - t_n\right\}\phi(\mathbf{x}_n) = \sum_{n=1}^{N} a_n\phi(\mathbf{x}_n) = \mathbf{\Phi}^{\mathrm{T}}\mathbf{a}$$

RÉGRESSION

Sujets: régression

- Donc, la solution w est simplement une somme pondérée des entrées $\mathbf{x}_n$ dans l'ensemble d'entraînement

$$\mathbf{w} = \sum_{n=1}^{N} a_n \boldsymbol{\phi}(\mathbf{x}_n) = \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a}$$

$$\boxed{\mathbf{a} = (a_1, \ldots, a_N)^{\mathrm{T}}}$$

où chaque a_n est la contribution de $\phi(\mathbf{x}_n)$ à la solution

- **Idée :** plutôt qu'optimiser par rapport à $\mathbf{w}$, optimisons par rapport à $\mathbf{a}$

REPRÉSENTATION DUALE

Sujets: représentation duale

- Si on remplace w par $\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a}$, on peut démontrer qu'on obtient

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a}$$

- C'est la **représentation duale** de $J(\mathrm{w})$

REPRÉSENTATION DUALE

Sujets: représentation duale, matrice de Gram

- Si on remplace w par $\mathbf{\Phi}^{\mathrm{T}}\mathbf{a}$, on peut démontrer qu'on obtient

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{K}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{a}$$

- On va aussi noter $\phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m)$
- La **matrice de Gram** K contient tous les

$$K_{nm} = k(\mathbf{x}_n, \mathbf{x}_m)$$

de l'ensemble d'entraînement

Représentation Duale

Sujets: représentation duale

- Si on remplace $\mathbf{w}$ par $\mathbf{\Phi}^{\mathrm{T}}\mathbf{a}$, on peut démontrer qu'on obtient

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{K}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{a}$$

- En fixant à 0 les gradients par rapport à $\mathbf{a}$, on obtient

$$\mathbf{a} = \left(\mathbf{K} + \lambda\mathbf{I}_N\right)^{-1}\mathbf{t}$$

REPRÉSENTATION DUALE

Sujets: représentation duale

- Pour faire une prédiction :

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) = \mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\phi(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}}\left(\mathbf{K} + \lambda\mathbf{I}_N\right)^{-1}\mathbf{t}$$

où $\mathbf{k}(\mathbf{x}) = (k(\mathbf{x}_1, \mathbf{x}), \ldots, k(\mathbf{x}_N, \mathbf{x}))^{\mathrm{T}}$

RÉGRESSION À NOYAU

Sujets: régression à noyau

- Algorithme de régression à noyau
  - entraînement : $\mathbf{a} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{t}$
  - prédiction : $y(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}} \mathbf{a}$
- Pour exécuter cet algorithme, on a seulement besoin de calculer les produits scalaires du noyau

$$\phi(\mathbf{x}_n)^{\mathrm{T}} \phi(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m)$$

- Par contre, on doit toujours avoir accès aux entrées de l'ensemble d'entraînement

Apprentissage automatique

Méthodes à noyaux - astuce du noyau

RÉGRESSION À NOYAU

Sujets: régression à noyau

- Algorithme de régression à noyau
  - entraînement : $\mathbf{a} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{t}$
  - prédiction : $y(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}} \mathbf{a}$

- Pour exécuter cet algorithme, on a seulement besoin de calculer les produits scalaires du noyau

$$\phi(\mathbf{x}_n)^{\mathrm{T}} \phi(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m)$$

ASTUCE DU NOYAU

Sujets: astuce du noyau

- L'**astuce du noyau** vise à exploiter cet observation
  - peut-on définir des noyaux tels que calculer $k(\mathbf{x}_n,\mathbf{x}_m)$ est plus efficace que de calculer $\phi(\mathbf{x}_n)$ et $\phi(\mathbf{x}_m)$ et ensuite faire $\phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m)$?

Astuce du Noyau

Sujets: astuce du noyau

- La réponse est oui !
- Exemple (D=2):

3 multiplications et 1 addition

$$
\begin{aligned}
k(\mathbf{x},\mathbf{z}) &= (\mathbf{x}^{\mathrm{T}}\mathbf{z})^2 = (x_1z_1 + x_2z_2)^2 \\
&= x_1^2z_1^2 + 2x_1z_1x_2z_2 + x_2^2z_2^2 \\
&= (x_1^2, \sqrt{2}x_1x_2, x_2^2)(z_1^2, \sqrt{2}z_1z_2, z_2^2)^{\mathrm{T}} \\
&= \phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{z})
\end{aligned}
$$

2 fois 4 multiplications (construire $\phi(\mathbf{x})$ et $\phi(\mathbf{z})$)
suivi de 3 multiplications et 2 additions (produit scalaire)

NOYAU POLYNOMIAL

Sujets: noyau polynomial

- Une forme générale est le **noyau polynomial**

$$k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^{M}$$

où c est une constante >0

- On peut montrer que le $\phi(\mathbf{x})$ implicite contient tous les produits possibles entre au plus M éléments de $\mathbf{x}$

$\phi(\mathbf{x}) = (c_0,$

NOYAU POLYNOMIAL

Sujets: noyau polynomial

- Une forme générale est le **noyau polynomial**

$$k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^M$$

où c est une constante >0

- On peut montrer que le $\phi(\mathbf{x})$ implicite contient tous les produits possibles entre au plus M éléments de $\mathbf{x}$

$$\phi(\mathbf{x}) = (c_0, c_1 x_1, \ldots, c_D x_D,$$

Noyau Polynomial

Sujets: noyau polynomial

- Une forme générale est le **noyau polynomial**

$$k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^M$$

où c est une constante >0

- On peut montrer que le $\phi(\mathbf{x})$ implicite contient tous les produits possibles entre au plus M éléments de $\mathbf{x}$

$$\phi(\mathbf{x}) = (c_0, c_1 x_1, \ldots, c_D x_D, \\ c_{11} x_1^2, c_{12} x_1 x_2, \ldots,$$

Noyau Polynomial

Sujets: noyau polynomial

- Une forme générale est le **noyau polynomial**

$$k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^M$$

où c est une constante >0

- On peut montrer que le $\phi(\mathbf{x})$ implicite contient tous les produits possibles entre au plus M éléments de $\mathbf{x}$

$$\begin{aligned}\phi(\mathbf{x}) = (&c_0, c_1 x_1, \ldots, c_D x_D,\\ &c_{11}x_1^2, c_{12}x_1x_2, \ldots,\\ &c_{111}x_1^3, c_{112}x_1^2x_2, c_{123}x_1x_2x_3, \ldots)\end{aligned}$$

MALÉDICTION DE LA DIMENSIONNALITÉ

Sujets: nombre de paramètres

RAPPEL

- Notre modèle de régression aura plus de paramètres

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D} \sum_{j=1}^{D} w_{ij} x_i x_j + \sum_{i=1}^{D} \sum_{j=1}^{D} \sum_{k=1}^{D} w_{ijk} x_i x_j x_k$$

  - pour $M = 3$,on a maintenant $1 + D + D^2 + D^3$ paramètres

- De façon générale, augmente selon $O(D^M)$!
  - pour $D{=}100$, $M{=}3$, on a déjà plus d'un million de paramètres

MALÉDICTION DE LA DIMENSIONNALITÉ

Sujets: nombre de paramètres

RAPPEL

- Notre modèle de régression aura plus de paramètres

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D} \sum_{j=1}^{D} w_{ij} x_i x_j + \sum_{i=1}^{D} \sum_{j=1}^{D} \sum_{k=1}^{D} w_{ijk} x_i x_j x_k$$

  - ~~pour $M = 3$,on a maintenant $1 + D + D^2 + D^3$ paramètres~~

- ~~De façon générale, augmente selon $O(D^M)$!~~

  - ~~pour D=100, M=3, on a déjà plus d'un million de paramètres~~

Malédiction de la Dimensionnalité

Sujets: nombre de paramètres

RAPPEL

- Notre modèle de régression aura plus de paramètres

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D}\sum_{j=1}^{D} w_{ij} x_i x_j + \sum_{i=1}^{D}\sum_{j=1}^{D}\sum_{k=1}^{D} w_{ijk} x_i x_j x_k$$

  - ~~pour $M = 3$,on a maintenant $1 + D + D^2 + D^3$ paramètres~~

On n'a plus à apprendre un paramètre w explicitement !

- ~~De façon gé~~

  - ~~pour D=100, M=3, on a déjà plus d'un million de paramètres~~

MALÉDICTION DE LA DIMENSIONNALITÉ

Sujets: malédiction de la dimensionnalité

RAPPEL

- La difficulté à bien généraliser peut donc potentiellement augmenter **exponentiellement** avec la dimensionnalité D des entrées

- Cette observation est appelée la **malédiction de la dimensionnalité**

- Nécessite le design de modèles / algorithmes appropriés pour chaque problème
  - on cherche des modèles / algorithmes qui vont bien exploiter les données à notre disposition

MALÉDICTION DE LA DIMENSIONNALITÉ

Sujets: malédiction de la dimensionnalité

RAPPEL

- La difficulté à bien généraliser peut donc potentiellement augmenter **exponentiellement** avec la dimensionnalité D des entrées

- (

On risque quand même d'être victime de sur-apprentissage, lorsque la dimensionnalité (implicite) de $\phi(\mathbf{x})$ augmente

- Nécessite le design de modèles / algorithmes appropriés pour chaque problème
  - on cherche des modèles / algorithmes qui vont bien exploiter les données à notre disposition

Apprentissage automatique

Méthodes à noyaux - construction de noyaux

Astuce du Noyau

Sujets: astuce du noyau

- L'**astuce du noyau** vise à exploiter cet observation
  - peut-on définir des noyaux tels que calculer $k(\mathbf{x}_n,\mathbf{x}_m)$ est plus efficace que de calculer $\phi(\mathbf{x}_n)$ et $\phi(\mathbf{x}_m)$ et ensuite faire $\phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m)$?

Noyau Polynomial

Sujets: noyau polynomial

- Une forme générale est le **noyau polynomial**

$$k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^{M}$$

où c est une constante >0

- On peut montrer que le $\phi(\mathbf{x})$ implicite contient tous les produits possibles entre au plus M éléments de $\mathbf{x}$

$$\begin{aligned}\phi(\mathbf{x}) = (&c_0,\ c_1 x_1, \ldots, c_D x_D, \\ &c_{11} x_1^2, c_{12} x_1 x_2, \ldots, \\ &c_{111} x_1^3, c_{112} x_1^2 x_2, c_{123} x_1 x_2 x_3, \ldots)\end{aligned}$$

CONSTRUCTION DE NOYAUX

Sujets: construction de noyau

- Règles pour **construire de nouveaux noyaux** valides

$$
\begin{aligned}
k(\mathbf{x},\mathbf{x}') &= c k_1(\mathbf{x},\mathbf{x}') \\
k(\mathbf{x},\mathbf{x}') &= f(\mathbf{x}) k_1(\mathbf{x},\mathbf{x}') f(\mathbf{x}') \\
k(\mathbf{x},\mathbf{x}') &= q\left(k_1(\mathbf{x},\mathbf{x}')\right) \\
k(\mathbf{x},\mathbf{x}') &= \exp\left(k_1(\mathbf{x},\mathbf{x}')\right) \\
k(\mathbf{x},\mathbf{x}') &= k_1(\mathbf{x},\mathbf{x}') + k_2(\mathbf{x},\mathbf{x}')
\end{aligned}
$$

$$
\begin{aligned}
k(\mathbf{x},\mathbf{x}') &= k_1(\mathbf{x},\mathbf{x}') k_2(\mathbf{x},\mathbf{x}') \\
k(\mathbf{x},\mathbf{x}') &= k_3\left(\boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}')\right) \\
k(\mathbf{x},\mathbf{x}') &= \mathbf{x}^{\mathrm{T}} \mathbf{A} \mathbf{x}' \\
k(\mathbf{x},\mathbf{x}') &= k_a(\mathbf{x}_a,\mathbf{x}'_a) + k_b(\mathbf{x}_b,\mathbf{x}'_b) \\
k(\mathbf{x},\mathbf{x}') &= k_a(\mathbf{x}_a,\mathbf{x}'_a) k_b(\mathbf{x}_b,\mathbf{x}'_b)
\end{aligned}
$$

où $c>0$, $f(\mathbf{x})$ est une fonction, $q(a)$ est un polynôme avec coefficients positifs, $\mathbf{A}$ est une matrice définie positive et $\mathbf{x}=(\mathbf{x}_a,\mathbf{x}_b)$.

Les noyaux k_1, k_2, k_3, k_a et k_b **doivent être valides**.

CONSTRUCTION DE NOYAUX

Sujets: construction de noyau

- Exemple : $ck_1(\mathbf{x}, \mathbf{x}')$ où $c{>}0$

$$\begin{aligned} ck_1(\mathbf{x}, \mathbf{x}') &= c\phi^1(\mathbf{x})^{\mathrm{T}}\phi^1(\mathbf{x}') \\ &= \left(\sqrt{c}\phi^1(\mathbf{x})\right)^{\mathrm{T}}\left(\sqrt{c}\phi^1(\mathbf{x}')\right) \\ &= \phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{x}') \end{aligned}$$

Noyau gaussien

Sujets: noyau gaussien

- Un noyau souvent utilisé est le **noyau gaussien** :

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2\right)$$

- Est valide puisque :

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{x} + (\mathbf{x}')^{\mathrm{T}}\mathbf{x}' - 2\mathbf{x}^{\mathrm{T}}\mathbf{x}'$$

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\mathbf{x}^{\mathrm{T}}\mathbf{x}/2\sigma^2\right) \exp\left(\mathbf{x}^{\mathrm{T}}\mathbf{x}'/\sigma^2\right) \exp\left(-(\mathbf{x}')^{\mathrm{T}}\mathbf{x}'/2\sigma^2\right)$$

NOYAU GAUSSIEN

Sujets: noyau gaussien

- Un noyau souvent utilisé est le **noyau gaussien** :

$$k(\mathbf{x},\mathbf{x}') = \exp\left(-\|\mathbf{x}-\mathbf{x}'\|^2/2\sigma^2\right)$$

$$k(\mathbf{x},\mathbf{x}') \quad = \quad c k_1(\mathbf{x},\mathbf{x}')$$

- Est valide puisque :

$$\|\mathbf{x}-\mathbf{x}'\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{x} + (\mathbf{x}')^{\mathrm{T}}\mathbf{x}' - 2\mathbf{x}^{\mathrm{T}}\mathbf{x}'$$

$$k(\mathbf{x},\mathbf{x}') = \exp\left(-\mathbf{x}^{\mathrm{T}}\mathbf{x}/2\sigma^2\right)\exp\left(\overbrace{\mathbf{x}^{\mathrm{T}}\mathbf{x}'/\sigma^2}\right)\exp\left(-(\mathbf{x}')^{\mathrm{T}}\mathbf{x}'/2\sigma^2\right)$$

NOYAU GAUSSIEN

Sujets: noyau gaussien

- Un noyau souvent utilisé est le **noyau gaussien** :

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2\right)$$

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= c k_1(\mathbf{x}, \mathbf{x}') \\ k(\mathbf{x}, \mathbf{x}') &= \exp\left(k_1(\mathbf{x}, \mathbf{x}')\right) \end{aligned}$$

- Est valide puisque :

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{x} + (\mathbf{x}')^{\mathrm{T}}\mathbf{x}' - 2\mathbf{x}^{\mathrm{T}}\mathbf{x}'$$

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\mathbf{x}^{\mathrm{T}}\mathbf{x}/2\sigma^2\right) \overbrace{\exp\left(\overbrace{\mathbf{x}^{\mathrm{T}}\mathbf{x}'/\sigma^2}\right)} \exp\left(-(\mathbf{x}')^{\mathrm{T}}\mathbf{x}'/2\sigma^2\right)$$

NOYAU GAUSSIEN

Sujets: noyau gaussien

- Un noyau souvent utilisé est le **noyau gaussien** :

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2\right)$$

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= c k_1(\mathbf{x}, \mathbf{x}') \\ k(\mathbf{x}, \mathbf{x}') &= \exp\left(k_1(\mathbf{x}, \mathbf{x}')\right) \\ k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x}) k_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}') \end{aligned}$$

- Est valide puisque :

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{x} + (\mathbf{x}')^{\mathrm{T}}\mathbf{x}' - 2\mathbf{x}^{\mathrm{T}}\mathbf{x}'$$

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\mathbf{x}^{\mathrm{T}}\mathbf{x}/2\sigma^2\right) \exp\left(\mathbf{x}^{\mathrm{T}}\mathbf{x}'/\sigma^2\right) \exp\left(-(\mathbf{x}')^{\mathrm{T}}\mathbf{x}'/2\sigma^2\right)$$

NOYAU GAUSSIEN

Sujets: noyau gaussien

- Un noyau souvent utilisé est le **noyau gaussien** :

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2\right)$$

- On peut même montrer que le $\phi(\mathbf{x})$ est un vecteur de taille infinie !

CONSTRUCTION DE NOYAUX

Sujets: construction de noyaux

- On peut également définir des noyaux pour des entrées qui ne sont pas des vecteurs $\mathbf{x}$ de taille fixe
  - chaînes de caractères
  - ensembles de vecteurs
  - etc.

- Noyau de *Fisher* : un paradigme pour dériver de nouveaux noyaux à partir de modèles probabilistes génératifs
  - voir fin de la section 6.2

Apprentissage automatique

Méthodes à noyaux - résumé

RÉGRESSION À NOYAU

Sujets: résumé de la régression à noyau

- Modèle : $y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}}\mathbf{a} = \sum_{n=1}^{N} a_n k(\mathbf{x}, \mathbf{x}_n)$

  $$p(t|\mathbf{x}, \beta) = \mathcal{N}(t|y(\mathbf{x}), \beta^{-1})$$

- Entraînement : $\mathbf{a} = (\mathbf{K} + \lambda\mathbf{I}_N)^{-1}\,\mathbf{t}$
- Hyper-paramètres : λ et ceux dans le noyau $k(\mathbf{x}, \mathbf{x}_n)$
  - c et M pour le noyau polynomial
  - σ^2 pour le noyau gaussien
- Prédiction : $y(\mathbf{x})$

Capacité et Noyau

Sujets: lien entre capacité et noyau

- Modèle : $y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}}\mathbf{a} = \sum_{n=1}^{N} a_n k(\mathbf{x}, \mathbf{x}_n)$
- Noyau polynomial $k(\mathbf{x}, \mathbf{x}') = \left(\mathbf{x}^{\mathrm{T}}\mathbf{x}' + c\right)^M$
  - plus M est grand, plus le modèle a de la capacité
- Noyau gaussien $k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2\right)$
  - plus σ^2 est petit, plus le modèle a de la capacité